DOCKET NO.: HITACHI-0020 **PATENT**

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re patent application of: Nobuyuki YAMASHITA

Serial No.: Unknown Group No.: Unknown

Filed: Unknown Examiner: Unknown

For: **METHOD AND SYSTEM FOR MULTIDIMENTIONAL DATABASE MANAGEMENT**

I, Ken I. Yoshida, Registration No. 37,009 certify that this correspondence is being deposited with the U.S. Postal Service as First Class mail in an envelope addressed to the Assistant Commissioner for Patents, Washington, D.C. 20231.

On Dec 13, 2001

Ken I. Yoshida Reg. No. 37,009

Assistant Commissioner for Patents
Washington, D.C. 20231

PRELIMINARY AMENDMENT

Sir:

Please make the following change prior to examination of the above-referenced application:

In the Specification:

Please amend the following:

Line 9, page 1, please change “layer” to –layer or hierarchy--.

Line 2, page 2, please change “unit dimension” to –unit or measure dimension--.

Line 6, page 2, please change “layer” to –layer or hierarchy --.

Line 7, page 2, please change “layer” to –layer or hierarchy --.

Line 6, page 3, please change “separated by lines” to –separated by new line characters --.

Line 14, page 4, please change "summarize" to --aggregate--.

Line 15, page 4, please change "22461" to --224261--.

Line 1, page 5, please change "refrigerator" to --PCs--.

Line 26, page 10, between "the current invention." and "The multidimensional", please insert – In the specification of the current application, the term, "layer" is interchangeably used with the term, "hierarchy" or "hierarchical" to have the substantially identical meaning. Similarly, the terms, "character row conversion" and "character string replacement" are interchangeably used in the current application.--

Line 28, page 10, please change "main memory" to --main or primary memory--.

Line 1, page 11, please change both occurrences of "layer" to --layer or hierarchy --.

Line 2, page 11, please change both occurrences of "layer" to --layer or hierarchy --.

Line 24, page 16, please change the second occurrence of "3212" to --3213--.

Lines 2, page 21, please change "the dimensional layer information unit 41" to -- dimensional layer information 41--.

Line 3, page 21, please change "dimensional layer information unit 41" to -- dimensional layer information 41--.

Lines 11, page 21, please change "RDB Product Master List" to --Product Master Table--.

Line 11, page 21, please change "fourth and the fifth" to --fifth and the sixth--.

Line 6, page 22, please change “the layer information is not to be generated from the RDB 26” to –the hierarchy rule 42 is not defined that the hierarchical information is to be generated from RDB 26--.

Line 8, page 22, please change “members” to –member--.

Line 10, page 22, please change “information” to –information of the specified member--.

In the Drawings:

In FIGURES 8 and 26, please change every occurrence of “¥” to - -\--.

In FIGURE 11, please change “CSV FILE” to –CSV FILE1--.

In FIGURE 15, please change both occurrences of “RDB PRODUCT MASTER LIST” to –RDB TABLE ‘PRODUCT MASTER TABLE’--.

In FIGURE 17, please change in the step 9023, “BASED ON SPECIFIED PARAMETERS AND MEMBERS OF LAYER RULE” to –WITH THE SPECIFIED MEMBER AND PARAMETERS IN HIERARCHY RULE--.

Respectfully submitted,

Ken I. Yoshida, Esq.
Reg. No. 37,009

Date: December 17, 2001

KNOBLE & YOSHIDA LLC
Eight Penn Center, Suite 1350
1628 John F. Kennedy Blvd.
Philadelphia, PA 19103
(215) 599-0600

METHOD AND SYSTEM FOR MULTIDIMENTIONAL DATABASE MANAGEMENT

This is a continuation of prior application Serial No. 09/994,950 filed on November 27, 2001 under 35 C.F.R. 1.53(b)

Field of the Invention

The current invention is generally related to multidimensional database management, and more particularly related to processing multidimensional data without ~~layer~~ layer or hierarchy structure information.

BACKGROUND OF THE INVENTION

A multidimensional database model contains data corresponding to a point having values in multidimensional space that is defined by a plurality of dimensions. The multidimensional database model is generally effective in analyzing the data in a multifaceted manner. Based upon a predetermined rule, it is a basic function to correspond a value of data that corresponds to one or more of points in the multidimensional space to other values corresponding to other points. For example, the above basic function is disclosed in "OLAP Practical Data Warehouse," Toyoshima and Kimura, pp76-79, (1997); "OLAP Solutions Building Multidimensional Information Systems," Thomsen, pp. 89-104, (1997). The above described basic function is useful in constructing a database with minimally necessary data input as well as in retrieving data that has been already calculated.

In a multidimensional database model, the multidimensional space as defined by a plurality of dimensions is called "cube." The dimensions in the cube is generally expressed by a set of members that have a layer structure. A point in the multidimensional space as defined by an arbitrary member at each dimension is named "cell," and a value of

the data corresponding to the cell is “a cell value.” For example, using multidimensional data that represents sales units and sales amounts of products sold by a company A in Japan, the cube includes a time dimension, a retail store dimension, a merchandise dimension and a ~~unit dimension~~unit or measure dimension.

To illustrate the above related example, FIGURE 20 is a diagram for describing conventional layer structure information storing layer information on members for each dimension. A ~~layer~~ layer or hierarchy 4301 for the time dimension has “1999” as the highest member or the highest member in the ~~layer~~ layer or hierarchy structure, and the highest member 1999 in turn has two child members, 1999Q1 and 1999Q2. Similarly, the child member 1999Q1 has grand child members, 199901, 199902 and 199903. By the same token, the other child member 1999Q2 has grand child members, 199904 and 199905. The grand children members, 199901, 199902, 199903, 199904 and 199905 do not have any great grand children and are defined as the lowest members in the above example. The level is accordingly assigned to the members based upon a position in the layer. The lowest members are situated at Level 0 while the direct parent of the lowest members is located at Level 1. Similarly, the direct parent of Level 1 members are located at Level 2. The highest members are located at the level that is equal to the number of layers minus one.

Still referring to FIGURE 20, a layer 4302 for a retail store dimension has all the sales territories or nation wide territories as the highest member. The highest member has two child members including Eastern Japan and Western Japan. Eastern Japan further owns two child members including Chiba store and Saitama store. Western Japan further owns two child members including Osaka store and Hiroshima store. Similarly, a layer 4303 for a merchandise dimension has the all merchandises as the highest member. The highest member has two child members including home appliance and audiovisual equipment (AV). Home Appliance further owns two child members including washers and refrigerators. AV further owns two child members including television sets and video equipment. Lastly, a layer 4304 in a unit dimension has two members including a number

of sales and an amount of sales but, the members in the unit dimension have no layer structure.

Now referring to FIGURE 21, a diagram illustrates an example of conventional layer structure definition data. The conventional layer structure definition data 3201 through 3204 respectively define the time dimension, the retail store dimension, the merchandise dimension and the unit dimension, and each of the conventional layer structure definition data 3201 through 3204 contains a plurality of records that are ~~separated by lines~~separated by new line characters. Each of the records has members that are separated by commas, and the separation format is called comma separated value format (CSV). One record potentially includes all the members from the highest member to the lowest member. The conventional layer structure definition data 3201 through 3203 each is organized to list records according to the level. That is, in the above example, a record at Level 2 is followed by a record at Level 1 and then by a record at Level 0. For the layer structure definition data 3204, a record has only members at Level 0.

Now referring to FIGURE 22, a diagram illustrates an example of conventional data. The data 3301 is stored in the cube and in the above described CSV format. Each of the record includes members and corresponding cell values. In the example, each record thus contains five fields respectively for information on month/year, a retail store, a merchandise name, a number of sales and an amount of sales. The first three fields are respectively from the time dimension, the retail store dimension and the merchandise dimension. On the other hand, the last two fields are cell values representing the number of sales and the amount of sales from the unit dimension.

FIGURE 23 is a diagram illustrating an exemplary display of multidimensional data. The exemplary display is a screen multidimensional data analysis on a terminal device. The horizontal axis includes members on the time dimension while the vertical axis includes members on the retail store dimension. All members from the above dimensions are displayed. On the other hand, as shown in the upper right corner of the

of sales and an amount of sales but, the members in the unit dimension have no layer structure.

Now referring to FIGURE 21, a diagram illustrates an example of conventional layer structure definition data. The conventional layer structure definition data 3201 through 3204 respectively define the time dimension, the retail store dimension, the merchandise dimension and the unit dimension, and each of the conventional layer structure definition data 3201 through 3204 contains a plurality of records that are ~~separated by lines~~separated by new line characters. Each of the records has members that are separated by commas, and the separation format is called comma separated value format (CSV). One record potentially includes all the members from the highest member to the lowest member. The conventional layer structure definition data 3201 through 3203 each is organized to list records according to the level. That is, in the above example, a record at Level 2 is followed by a record at Level 1 and then by a record at Level 0. For the layer structure definition data 3204, a record has only members at Level 0.

Now referring to FIGURE 22, a diagram illustrates an example of conventional data. The data 3301 is stored in the cube and in the above described CSV format. Each of the record includes members and corresponding cell values. In the example, each record thus contains five fields respectively for information on month/year, a retail store, a merchandise name, a number of sales and an amount of sales. The first three fields are respectively from the time dimension, the retail store dimension and the merchandise dimension. On the other hand, the last two fields are cell values representing the number of sales and the amount of sales from the unit dimension.

FIGURE 23 is a diagram illustrating an exemplary display of multidimensional data. The exemplary display is a screen multidimensional data analysis on a terminal device. The horizontal axis includes members on the time dimension while the vertical axis includes members on the retail store dimension. All members from the above dimensions are displayed. On the other hand, as shown in the upper right corner of the

member 3311 is unregistered in the time dimensional layer 4301 as shown in FIGURE 20. Similarly, a second field of a second record is a member 3312 that has a value of "Yokohama Store." The member 3312 is unregistered in the retail store dimensional layer 4302 as shown in FIGURE 20. Lastly, a third field of a third record is a member 3313 that has a value of "~~refrigerator~~PCs." The member 3313 is unregistered in the merchandise dimensional layer 4303 as shown in FIGURE 20. In general, there are two ways to process the above described unregistered member data. The first way is to regard the unregistered member data to be invalid, and its corresponding record that contains the unregistered member data is also regarded as being invalid. The second way is to regard the unregistered member data to be new.

The above two options are further related to the database operations. The first processing option terminates the data handling operation upon detecting a record containing any unregistered member data. Alternatively, the data handling operation skips the record to a next one according to the first processing option. The data handling operation terminates by issuing an alarm signal, and the unregistered member remains to be unregistered at the layer information. Of course, the corresponding record is not included in the database and remains excluded from analysis. On the other hand, the above second processing option registers the currently unregistered member. In registering the new member, there are two ways to find a new position in the layer. One position is created at a new location as a new member without having any relation to the existing members.

Now referring to FIGURE 25, a diagram illustrates layer structure information for conventional data that includes members that are not previously registered in the layer structure information. The layer 4305 indicates that an unregistered member, "Yokohama store" is now registered as an independent member 4306 in the layer information structure 4305. The above described processing allows the incorporation of the unregistered member 4306 into the layer of the multidimensional database. Despite the incorporation of the previously unregistered member 4306, during an analysis stage, since the newly

system according to the current invention. In the specification of the current application, the term, "layer" is interchangeably used with the term, "hierarchy" or "hierarchical" to have the substantially identical meaning. Similarly, the terms, "character row conversion" and "character string replacement" are interchangeably used in the current applicaton. The multidimensional database processing system includes a computer system 1 that includes a central processing unit (CPU) 2, a ~~main memory~~main or primary memory unit 3, a secondary or external memory unit 4 such as magnetic disks and a plurality of terminals 6 that are connected by a network 5. A multidimensional database management unit 10 includes a system control unit 11, a multidimensional data management unit 12, a dimensional ~~layer~~ layer or hierarchy information management unit 13, a ~~layer~~ layer or hierarchy rule management unit 14, a ~~layer~~ layer or hierarchy structure information management unit 15 and a ~~layer~~ layer or hierarchy structure information update unit 16. The multidimensional database management unit 10 exists in the main memory unit 3 and includes the system control unit 11, the multidimensional data management unit 12 and the dimensional layer information management unit 13. The system control unit 11 controls the system in general. In particular, the system control 11 receives a layer rule definition request 21, a multidimensional data storage request 22 and a multidimensional data analysis request 23 and sends the requests 21, 22 and 23 to the multidimensional data management unit 12 and the dimensional layer information management unit 13. The system control 11 subsequently sends the processed results back to the requested terminals 6. The multidimensional data management unit 12 generally manages a multidimensional database 44 in an external or secondary memory unit 4. The multidimensional data management unit 12 has a first function to read storage data 33 and to store it in the multidimensional database 44, a second function to search in the multidimensional database 44 and to return the multidimensional data as well as a third function to return the data that is specified for an analysis by the search range in the multidimensional database 44.

Still referring to FIGURE 1, the dimensional layer information management unit 13 further includes the layer rule management unit 14, the layer structure information

definition data 3110 as shown in FIGURE 8, the layer structure definition data 3211 is generated as layer information. Layer rule definition data for generating the layer structure definition data 3212 or ~~3212~~ 3213will be described later.

FIGURE 10 is a diagram illustrating an example of the layer structure information 43 after being modified by a preferred embodiment according to the current invention. Layers 4311 through 4313 show the layer structure information 43 after the layer information of the unregistered members 3311 through 3313 of FIGURE 24 is registered using the layer structure definition data 3211 through 3213 of FIGURE 9. For example, the layer 4301 as shown in FIGURE 20 is registered at the layer structure information 43 of FIGURE 1 and the layer rule definition data 3110 as illustrated in FIGURE 8 is registered in the layer rule 42 of FIGURE 1. When the member 3311 of the first record in the layer rule definition data 3110 is to be stored in the multidimensional database 44 of FIGURE 1, it is detected that the member 3311 is unregistered. Upon the detection, the layer rule definition data 3110 is used to generate the layer structure definition data 3211 as shown in FIGURE 9 for the above unregistered member, and the newly generated layer structure definition data 3211 is stored at the layer structure information 43. As a result, the time dimensional layer 4301 as shown in FIGURE 20 changes to the time dimensional layer 4311 as shown in FIGURE 10. As described above, the preferred embodiment according to the current invention generates the layer information for a specified member by converting a character row of a member name according to a predetermined formal expression. As illustrated in the example of the time dimension, the member name information is efficiently generated according to a predetermined rule and is stored at the layer structure information. Since the layer rule determines the layer information generation, the multidimensional database improves the management efficiency.

A second preferred embodiment of the multidimensional database system according to the current invention generates the layer information for an unregistered member based upon information obtained from the CSV formatted file. The second

generated for an unregistered member and is stored at the layer structure information 43. In general, the above process for the third preferred embodiment is substantially identical to that of the first preferred embodiment except for the layer structure information update process 90B as shown in FIGURE 17.

Now referring to FIGURE 14, a diagram illustrates the third preferred embodiment of the multidimensional database system according to the current invention. In the third preferred embodiment, a multidimensional database management unit 10 is connected to a RDB management system 25 and has access to a RDB 26 via a predetermined set of procedures. In the third preferred embodiment, the multidimensional database management unit 10 is further connected to the dimensional layer information ~~unit~~ 41 and the multidimensional database 44 as shown in the first preferred embodiment with respect to FIGURE 1. The dimensional layer information ~~unit~~ 41 further includes the layer rule 42 as well as the layer structure information 43.

Now referring to FIGURE 15, a diagram illustrates a portion of merchandise dimensional layer rule definition data 3130 that is used in the third preferred embodiment according to the current invention. The exemplary layer rule 42 includes the above layer rule definition data 3130. The first two lines in the layer rule definition data 3130 are comments. The third line indicates that the layer information is to be generated from a data table named "~~RDB-Product-Master-List.~~Product Master Table." The fourth and fifth lines respectively indicate the correspondence of the LEVEL 0 members and LEVEL 1 members in the above specified list. The sixth line indicates that the LEVEL 2 member is "All Products" or "All Merchandise."

Now referring to FIGURE 16, a diagram illustrates an exemplary RDB product master table for the RDB that is used in the third preferred embodiment according to the current invention. The RDB product master table stores member names on the merchandise dimension in the RDB 26. In combination with the information on the fourth and the following lines in the layer rule definition data 3130 of FIGURE 15, the layer

information is generated for the merchandise dimension. For example, given an unregistered member, "PCs," the RDB 26 is searched to obtain a record 261 whose data value for the small classification is "PCs." The ~~fourth~~ fifth and the ~~fifth~~ sixth lines in the layer rule definition data 3130 of FIGURE 15 specify the LEVEL 1 and LEVEL 2 members that correspond to "computers" and "all merchandises."

Now referring to FIGURE 17, a flow chart illustrates steps involved in a third preferred process of the layer structure information updating process 90B according to the current invention. The layer rule 42 and the above specified member from the layer structure information management unit 15 are inputted to the layer structure information updating process 90B. In a step 9021, the layer structure information update unit 16 receives an instruction for layer information generation for the above specified member from the layer structure information management unit 15 and examines the layer rule 42. If ~~the layer information is not to be generated from the RDB 26~~the hierarchy rule 42 is not defined that the hierarchical information is to be generated from RDB 26, the third preferred process terminates. On the other hand, if the layer information is to be generated from the RDB 26, the third preferred process branches to a step 9022. In the step 9022, the RDB management system 25 is connected based upon a predetermined procedure. In a step 9023, an inquiry is made to the RDB 26 based upon the above specified ~~members~~ member and the parameters that are specified in the layer rule 42. In the step 9024, the layer structure definition data is generated as the layer ~~information~~ information of the specified member from the inquiry results according to the layer rule 42. In the step 9024, by using the generated layer structure definition data, the layer information of the specified member is also stored in the layer structure information 43, and the above generated layer information is returned to from the layer structure information management unit 15. The third preferred process then terminates.

As described above, in the third preferred process, the multidimensional database improves the management efficiency. For example, the layer 4303 as shown in FIGURE 20 is registered at the layer structure information 43 of FIGURE 1 and the retail store

FIG.8

TIME DIMENSIONAL LAYER RULE DEFINITION DATA

3110

```
#LAYER RULE :
#AFTER CHARACTER ROW CONVERSION ON MEMBER, OBTAINING LAYER INFORMATION

FROM MEMBER NAME
LEVEL10= $ MEMBER NAME
LEVEL11={
   S/¥(.... \ )0[123]/ \1Q1/
   S/¥(.... \ )0[456]/ \1Q2/
   S/¥(.... \ )0[789]/ \1Q3/
   S/¥(.... \ )1[012]/¥1Q4/
]
LEVEL12={
   S/\(.... \ )../\1/
]
```

FIG.26

RETAIL STORE DIMENSION LAYER DEFINITION DATA

```
#MAIN LAYER RULE:
#SEQUENTIALLY APPLYING SUBRULES
APPLYING SUBRULE STORE_RULE1
APPLYING SUBRULE STORE_RULE2
```
3140

```
#SUB LAYER RULE :STORE_RULE1
#FROM CSV FILE,OBTAINING LAYER INFORMATION
FROM FILE,CSV FILE1
LEVEL 0=COLUMN 3
LEVEL 1=COLUMN 2
LEVEL 2=COLUMN 1
```
3141

```
#SUB LAYER RULE:STORE_RULE 2:
#AFTER CHARACTER ROW CONVERSION ON RETAIL STORE, OBTAINING LAYER INFORMATION
FROM MEMBER NAME
LEVEL 0= $ MEMBER NAME
LEVEL1={
   S/\(.JAPAN\). $/\1/
}
LEVEL 2={
     ALL SALES TERRITORIES
}
```
3142

FIG.27

3340

```
199901,  EASTERN JAPAN  SAPPORO STORE, TVS,17, 1870000
199902,  HIROSHIMA STORE, VIDEOS, 42, 2940000
199903,  EASTERN JAPAN  SAPPORO STORE, REFRIGERATOR,15.1350000
```

FIG.11

3120

RETAIL STORE DIMENSIONAL LAYER RULE DEFINITION DATA

```
#LAYER RULE :
#OBTAINING LAYER INFORMATION FROM CSV FORMATTED FILE

FROM FILE,CSV FILE1
LEVEL0=COLUMN 3
LEVEL1=COLUMN 2
LEVEL2=COLUMN 1
```

FIG.12

3126

```
ALL AREAS,EASTERN JAPAN,CHIBA STORE
ALL AREAS,EASTERN JAPAN,SAITAMA STORE
ALL AREAS,EASTERN JAPAN,YOKOHAMA STORE
ALL AREAS,WESTERN JAPAN,OSAKA STORE
ALL AREAS,WESTERN JAPAN,HIROSHIMA STORE
```

FIG.15

3130

MERCHANDISE DIMENSIONAL LAYER RULE DEFINITION DATA

#LAYER RULE : RDB TABLE 'PRODUCT MASTER TABLE'
#OBTAINING FROM COL1,COL2
FROM RDB TABLE 'PRODUCT MASTER TABLE'
LEVEL 0=SMALL CLASSIFICATION
LEVEL 1=LARGE CLASSIFICATION
LEVEL 2="ALL MERCHANDISE"

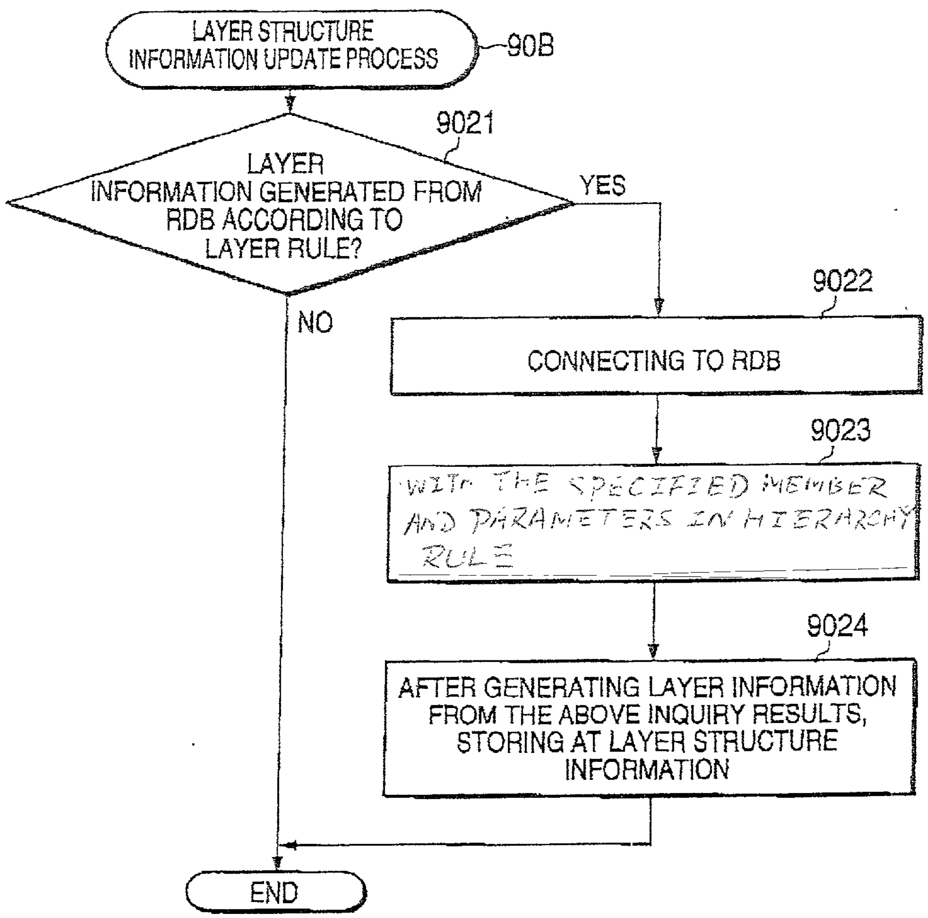
FIG.16

CONTENT OF RDB PRODUCT MASTER TABLE

LARGE CLASSIFICATION	SMALL CLASSIFICATION
HOME APPLIANCE	WASHERS
HOME APPLIANCE	REFRIGERATORS
AV	TVs
AV	VIDEOS
COMPUTERS	PCs

261

FIG.17


LAYER STRUCTURE INFORMATION UPDATE PROCESS
90B
9021
LAYER INFORMATION GENERATED FROM RDB ACCORDING TO LAYER RULE?
YES
NO
9022
CONNECTING TO RDB
9023
WITH THE SPECIFIED MEMBER AND PARAMETERS IN HIERARCHY RULE
9024
AFTER GENERATING LAYER INFORMATION FROM THE ABOVE INQUIRY RESULTS, STORING AT LAYER STRUCTURE INFORMATION
END